Blending History and Fiction in a Pervasive Game Prototype

Vlasios Kasapakis
Department of Cultural Technology and Communication
University of the Aegean
v.kasapakis@aegean.gr

Damianos Gavalas
Department of Cultural Technology and Communication
University of the Aegean
dgavalas@aegean.gr

ABSTRACT
Pervasive games represent an exciting development in gaming which leverages the use of sensor, visualization and networking technologies to provide immerse live-action game experiences. The field of pervasive games has been intensively researched in the recent years, as evidenced from the proliferation of available prototypes. Existing pervasive game projects commonly do not enable relocation of the game space while also overlooking several aspects which critically affect user acceptance and game experience such as scenario design, usability of employed technologies, game duration and intensity. This article introduces *Barbarossa*, an outdoor pervasive role-playing game. *Barbarossa* addresses the abovementioned issues featuring several portable game modes. It also takes into account concrete technology usage requirements for each game mode according to the game session duration and player effort required. Further, game experience is enhanced through incorporating several contextual parameters. User evaluation trials indicated warm reception of *Barbarossa* by players and confirmed that the main game design objectives have been largely achieved.

Categories and Subject Descriptors
D.3.3 [**Programming Languages**]: Language Constructs and Features – *abstract data types, polymorphism, control structures*.

General Terms
Algorithms, Performance, Design.

Keywords
Pervasive games; role-playing; orchestration; openness; user-generated content

1. INTRODUCTION
Pervasive games extend the gaming experience out into the real world be it on city streets, in the remote wilderness or a living room. Players equipped with mobile devices move through the game world, while sensors (either on-board or weaved into the game space) capture contextual information used to adapt game activities that vary depending on where users are, what they do or even how they feel [1, 2].

Pervasive games represent a rapidly growing field of research, as evidenced from the proliferation of available prototypes. Those prototypes may be classified in sub-genres mainly based on their supported features and game environment [2]. The most popular genres are *mobile* or *pure location-based games (*they take into account the player's changing relative or absolute position into the game rules) [3-10], *trans-reality games* (they incorporate physical, virtual and mixed reality game staging spaces) [11-15] and *augmented and mixed-reality (AR, MR)* games (they seek to integrate virtual and physical elements within a coherently experienced perceptual game world) [16-22].

Most existing pervasive games prototypes are location-aware, wherein location is commonly estimated by a GPS receiver [11, 13, 20] or GPS assisted by another technology, like computer vision) [23]. Occasionally, no GPS receiver is used and players denote their own location on a map [12]. As for networking, most of the games use GPRS technology [11, 12, 20] in addition to Bluetooth [11] or WLAN connectivity [20]. Some exclusively use WLAN connectivity [13]. Apart from location, most games also incorporate other context parameters like orientation [20, 23]. Further, many of the games take into account social context into their logic as the player is aware of other players' locations and activities [11-13, 20, 23].

The research issues typically dealt with in those games mainly related to the design of precise localization systems [9, 11, 13, 14, 20, 24], investigation of networking/communication options among players [6, 11, 12, 20, 23], design and usability of specialized player equipment [6, 9, 11, 17, 19, 20, 23], evaluation of usability and user experience [6, 11, 13, 17, 19, 20, 23, 25], etc. Many of those issues have been largely answered by recent technological developments (e.g. improved availability and precision outdoors positioning systems, advent of smartphones/tablets with sufficient capacity for high-speed Wi-Fi/3G networking and broad range of built-in sensors, decrease in cellular communication charges, etc). Further, several among the abovementioned pervasive game prototypes have been evaluated through user trials, conveying useful conclusions with regards to technical means to enhance players' immersion [6, 9, 11, 13, 14, 17, 19, 23-26]. However, the impact of pervasive technologies on the perception of the game world and the overall players' experience needs further investigation [27].

Notably, most prototyped pervasive games often fall short with respect to various design/gaming aspects. Firstly, the game content and staging is rigidly defined at design time by the game designer/moderator. Hence, it becomes difficult to relocate the game space elsewhere, even more so to engage the users in game

MUM '14, November 25 - 28 2014, Melbourne, VIC, Australia
Copyright 2014 ACM 978-1-4503-3304-7/14/11…$15.00
http://dx.doi.org/10.1145/2677972.2677981

design and content editing; those issues severely restrict games' portability, variability and openness [19], while players' involvement is less participatory. Secondly, pervasive games typically entail considerable orchestration resources; those may include dedicated infrastructure deployed throughout the game space in support of game activities or individuals/actors recruited to monitor players' conformation to game rules and/or provide assistance. Thirdly, several critical pervasive game aspects such as the design of the actual game scenarios, the usability of employed technologies, the game's duration and pace have been largely overlooked. Last, the fact that pervasive games are commonly played by players acting multiple, inter-dependent roles underlines the need to recruit highly motivated players, committed to undertake their game roles. We argue that an in-depth investigation of those aspects is of critical importance in order to substantiate game design decisions that will raise user acceptance and result in immersive game experiences.

In this paper we introduce *Barbarossa*, a trans-reality outdoors pervasive role-playing game, aiming at addressing the abovementioned shortcomings. Our focus is on designing an open and portable game engine architecture to ease game deployment at any setting, while relaxing the requirements for human support (i.e. orchestration) in the game setup.

The scenario of *Barbarossa* has been largely inspired by the medieval story of the pirate brothers Barbarossa, set in the Aegean Sea. Three player roles are identified in *Barbarossa*, each following an individual game sub scenario; their common objective is to trace and open a chest, locked with a four digit combination lock and hidden somewhere in the city of Mytilene (Lesvos, Greece), in order to claim the rewards kept inside it. The first player, the *Treasure Hunter*, has to locate the actual chest and the other two, the *Pirate* and the *Knight* have to find a part of the 4 digit combination code (divided into 2-digit code parts) so as to unlock the chest.

In Barbarossa, the implemented game scenarios are diverse with respect to their orchestration needs, technologies used, game session duration and effort required by the players. The reason of this diversification is, firstly, to cater for different user needs and preferences and, secondly, to serve as a testbed to assess the impact of different pervasive game modes on user perception, appreciativeness and experience. The main research contributions of our work are the following:

- The architecture of *Barbarossa* is designed so as to foster the active engagement of players in collaborative high-quality game content development, enhancing commitment, socialization and game awareness among players.
- We designed individual game modes able to function with few or no orchestration actions so that the player can play anywhere, anytime. Along with preventing user frustration this design approach eases the game setup [20, 28]; moreover, it allows users to interrupt the game with uncompleted tasks and resume later on, while it does not necessitate supervision by orchestration teams.
- We take into account the localization and communication uncertainty that emerges when using GPS, WLAN and GPRS. Along this line, we designed game modes with different localization and communication requirements, relevant to the game mode effort level (slow/fast pace).

The remainder of the article is structured as follows: Section 2 details the overall scenario as well as the individual sub-scenarios of *Barbarossa*. Section 3 discusses implementation issues and Section 4 reports the main findings drawn from our initial user evaluation trials. Finally, Section 5 concludes the article and draws directions for future work.

2. GAME SCENARIO

The plot of *Barbarossa* is inspired by the legendary adventures of the Barbarossa pirate brothers (Aruj, Ilyas and Hizir) who operated on the Aegean Sea, in the 16th century. Returning from a trading voyage to Tripoli, the boat of Aruj and Ilyas was captured and plundered by a galley from Rhodes Island, operated by the Knights of St. John. The Christians killed Ilyas and kept Aruj as a prisoner, until his family could arrange the payment of a release ransom [29].

The game scenario of Barbarossa involves a modern *Treasure Hunter* who discovers (within an old book) some information about a chest filled by the three Barbarossa brothers with their most valuable treasures. According to the book, the deal was that Hizir would decide where to hide the chest (somewhere in the city of Mytilene) while the other two brothers would lock it with a four digit combination lock setting 2 digits of the code each; the unlocking of the chest would require the presence of all the three brothers. Getting obsessed with the chest and desperate to find more clues, the *Treasure Hunter* performs a black magic ritual managing to contact the spirits of the *Knight* who wants to kill Ilyas and a *Pirate* member of Hizir Barbarossa's crew who is on his way to free Aruj. The *Treasure Hunter* commands them (on her dream) to learn about the code parts that unlock the chest, while she searches for the chest in the town of Mytilene. To fulfill the overall objective of *Barbarossa*, all three players should cooperate to discover and open a real/physical chest locked with a four digit combination lock.

In order to provide incentives to the players acting the three roles, *Barbarossa* defines three individual sub goals. The player who acts the *Treasure Hunter* has to locate the real chest, locked with the four digit combination lock. The four digit combination is broken down into two parts, comprising 2 random digits of the combination code each. The *Knight* and *Pirate* players follow and complete an individual sub scenario each that rewards each one with one part of the unlocking code.

The player enrolled in finding the chest acts the *Treasure Hunter* and pursues the *Treasure Hunt* scenario, trying to find the chest hidden by Hizir Barbarossa in the city of Mytilene. The player who acts the *Knight* pursues the *ManHunt* scenario to complete the game goal of obtaining her 2-digit part of the code. According to the *ManHunt* scenario, Ilyas managed to survive the battle and hid in a building, somewhere in the town of Mytilene. Guards situated at the docks of Mytilene noticed a suspicious vessel and suspect that Ilyas will try to embark and escape. The *Knight*'s objective, when pursuing the *ManHunt* scenario, is to approach and kill Ilyas chasing him along the city streets of Mytilene, so as to deter his escape.

The player acting the *Pirate* pursues the *Set Me Free!* scenario. According to the latter, following the battle among the Knights of St. John and the *Pirate*s, Aruj was captured by the Knights. They then engaged Aruj as a galley slave. However, their ship was damaged; hence, the Knights had to wait for another vessel to leave the island. To prevent anyone from attempting to free Aruj,

they decided to continuously move him around the city of Mytilene so no one could guess his exact location. They ordered two *Guards* to take turns in moving Aruj around and change shifts when needed. The *Guards* though, being passionate gamblers, started betting their shifts by rolling two dices. The looser guards Aruj for the next shift and decides about the next position that Aruj will be transferred to.

Apart from the *Pirate* player, the *Set Me Free!* scenario is supported by two other players acting the *Guards*. The *Guards* will take turns in rolling two virtual dices until someone achieves a 'double' roll and wins the first round. When a *Guard* 'picks up' the dices (in a custom Android application), she starts rolling for a specific amount of time (defined by an automated rule system, explained in detail in Section 5.3). The looser sets Aruj position into the real world using a custom Google Maps application and rolls the dices again until she gets a double roll to pass Aruj to the other player. Until then, the *Pirate* roams the city streets chasing an AR view of Aruj and his *Guard* trying to approach them, kill the *Guard* and free Aruj.

Upon the successful completion of the *ManHunt* and *Set Me Free!* scenarios, the system generates a message with the 2-digit code part rewarded to each player. All enrolled players need to complete their individual game goals successfully, as those are complementary in their common mission to unlock the chest and fulfill the overall game objective. The game sub scenarios are pursued individually. The players can play anywhere, anytime. In the *Set Me Free!* scenario, the *Pirate* player can invite players to act the *Guards* roles in order to acquire the required part of the code.

The players are introduced into the game, receiving a full manual of *Barbarossa* and using a smartphone device each, running separate custom applications, which support the individual scenarios. Certainly, a player may use her own smartphone, having downloaded and installed the respective application. The manual contains a QR-Code scanned by the *Treasure Hunter* in order to receive the first clue.

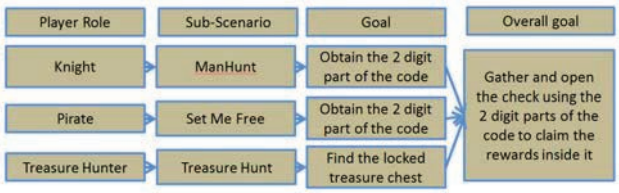

Fig. 1. The game structure of *Barbarossa*.

Upon completing their scenario, the players need to gather on the same physical location to unlock the chest found by the *Treasure Hunter*, using the code parts obtained by the two other players, so as to claim their rewards. The players will also have to cooperate to find the correct order of the digits which consists the lock combination in order to unlock the chest. Fig. 1 illustrates the *Barbarossa* phases, the player roles, the scenarios pursued and goals of each one as well as the overall goal of *Barbarossa*.

3. GAME MECHANICS AND RULES

3.1 The Treasure Hunter

The *Treasure Hunter* follows the *Treasure Hunt* scenario, which corresponds to prolonged time scale, requires restrained player effort, involves moderate usage of technology and requires some short of orchestration actions (QR-Codes placed at specific city locations, used to deliver clues) as well as the participation of recruited individuals along the game session.

The *Treasure Hunter* roams the cityscape scanning QR-Codes (using a custom Android application) questing the actual location of the treasure chest. For the QR-Code scanning, the ZXing barcode image processing library has been used [30]. When the player scans a QR-Code, she gets a clue that leads her to the next one. The clues typically vary from a short textual or visual description to telephone numbers or e-mail accounts of people that the player should communicate with to ask information about the next clue. Alternatively, clues may be obtained from SunSPOT sensor node devices [31] (located on specified spots) running a custom Java application: a hidden clue is revealed (visualized 'on the air' through the node's LED lights) when the player performs a specified gesture[1] (see Fig. 2), while the clue's content is periodically updated.

Fig. 2. Clue visualization through programmatically controlled LEDs of a SunSPOT device.

The player consumes energy units to retrieve clues. The consumption of energy units is proportional to the average environmental sound level recorded during the QR-Code scanning process. This is to encourage the *Treasure Hunter* to actually observe the environment around her and try to retrieve clues while remaining unnoticed. This way we aim to increase the immersion of the *Treasure Hunter*. The sound level is measured in decibels from a recording initiated when the ZXing application starts scanning the QR-Code until the moment that the QR-Code scanning is complete. To recharge energy points the *Treasure Hunter* has to switch to 'sleep' mode. The application then exits and, when resumed, it recharges one energy point for every minute it remained in 'sleep' mode. Fig. 3 presents the main interface of the *Treasure Hunt* application.

Fig. 3. The Treasure Hunt interface showing an example clue text pointing to a church nearby the city's castle, visualizing a church picture relevant to the clue text, the remaining energy points and the sound level of the last scan.

[1] Each SunSPOT device is equipped with an accelerometer as well as with LEDs which may be programmatically controlled (blinked). The player swings the SunSPOT from left to right, wide and fast. The device senses the acceleration and turns the LEDs on and off in such a way that the player reads words painted 'on the air' by the SunSPOT.

Fig. 4 presents the *Treasure Hunt* game functional scheme.

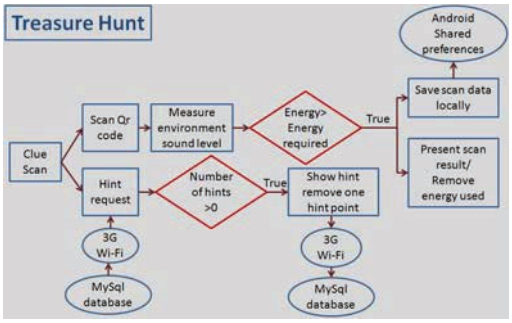

Fig. 4. The Treasure Hunt game functional scheme

3.2 The Knight

The *Knight* Player pursues the *ManHunt* scenario. *ManHunt* refers to a short time scale game session that requires intense player effort with increased use of technology. The *Knight* Player acts outdoors, operating a custom Android application on a device equipped with GPS receiver and 3G or Wi-Fi radio. The area around the port of Mytilene, where Ilyas will try to escape from, is visualized through a Google maps interface.

When a GPS position fix is acquired, the game starts as soon as the player presses the 'Start' button. Thereafter, a pirate marker (representing Ilyas) appears on a map interface, following a route starting at a random historical city location (his hideout) and ending at a randomly chosen point (among a list of points generated by parsing the results of a Google Places API [32] request for places nearby the port of Mytilene). The trajectory followed by the *Pirate* player is dictated by a Google Map Directions API [33] invocation. At the game startup, the system retrieves the current temperature on the *Knight*'s location (via the Yahoo Weather web service [34]) and uses it to adjust Ilyas speed. If the temperature is below 20^{o}C, Ilyas moves faster (the game session is shortened to protect the *Knight* from exposing herself to the cold). Likewise, between 20 ^{o}C and 30 ^{o}C Ilyas moves in normal speed, while above 30^{o}C in low speed.

The player (*Knight*) should approach Ilyas in a distance up to 60m so as to shoot him (by pressing a cannon button) and claim the part of the code kept by him. In order to be able to kill Ilyas, the *Knight* needs to have visual contact with him (this is inferred through a Google Maps Directions API invocation[2]). The player is also constantly informed of the distance between her and Ilyas as well as the distance between Ilyas and his escape point. When the game ends, a message is displayed revealing the 2-digit part of the combination code that unlocks the chest.

Fig. 5 presents the *ManHunt* application interface.

[2] A Google Maps Directions API function is invoked when the 'shoot' button is pressed to return directions from the *Knight*'s towards Ilyas current location. In case that the returned directions include 'turn' instructions, it is inferred that the Knight and Ilyas are not in visual contact.

Fig. 5. The ManHunt interface presenting the Ilyas Google Maps marker, the Knight's position, and the distance between Ilyas and Knight as well as the distance between Ilyas and his intended escape point.

Fig. 6 below illustrates the *ManHunt* game functional scheme.

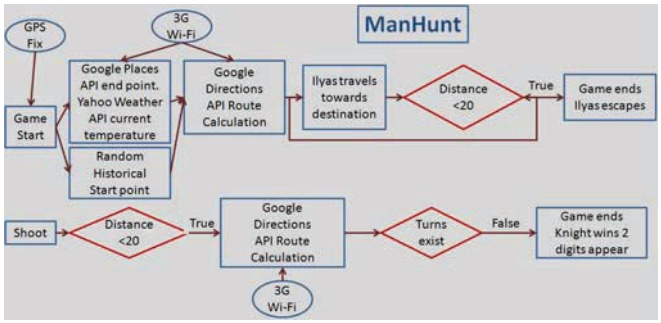

Fig. 6. The ManHunt game functional scheme.

3.3 The Pirate

Set Me Free! refers to a short timescale game session requiring intense player effort and utilizing a multitude of technologies including GPS, Wi-Fi/3G connection and AR. *Set Me Free!* enables a combination of a board game and an AR-based outdoor game. This game mode is self-orchestrated as the orchestration actions required are seamlessly embedded into the game process and undertaken by the players themselves having the form of player-generated content.

Upon the game startup, the *Pirate* (executing a custom Android application) appoints her physical location (GPS position fix) as the game's center. This is an orchestration action embedded into the game play, ensuring the game's portability (establishment of the playscape anywhere). The *Pirate* can launch the AR application (powered by the Android Augmented Reality Framework [35]) and tap the Center marker that appears inside her AR view to denote she is ready to play.

Fig. 7. The Set Me Free!'s Shift Adapter interface: Aruj is visualized as a guarded pirate marker, the last known position of the outdoors Pirate player is shown as a pirate marker, the center is shown as a castle marker and the virtual dices are displayed at the bottom right.

Having set the center, the *Guards* may enter the game. As discussed above, the *Guards* take turns in guarding Aruj and deciding his position by rolling two virtual dices. First, they need to execute a custom Android application called *Shift Adapter*. The application firstly adapts the game rules based on the actual *Pirate* player's age and the top speed of previous *Pirate* players of the same age (this is detailed later on in this section). It also instructs them to roll the dices once each. The *Guard* with the lowest roll looses and has to guard Aruj first. Namely, she is required to set Aruj position (through tapping on a Google Map interface) anywhere within a 100 meter radius from the center, as determined by the *Pirate* outdoors player. Fig. 7 illustrates the *Shift Adapter* application interface.

The Aruj AR marker (the position of Aruj is set by the *Guards*) then appears on the outdoor *Pirate* player *Set Me Free!* interface; the *Pirate* should then chase Aruj and kill the guard by tapping the AR marker when she approaches him in distance up to 15m. The brightness of the screen is automatically adjusted according to the environmental light intensity to deliver better AR content viewing conditions to the player. The setting of Aruj's outdoor position represents an orchestration action seamlessly embedded into the game play and game scenario.

Following the initial setting of Aruj on the map, the *Guard* who lost starts rolling the dices. When she scores a double roll, the other *Guard* takes turn in setting the next position for Aruj and starts rolling the dices again. The goal of the game for each of the *Guards* is to avoid being the one that guards Aruj at the time that the outdoor *Pirate* approaches and kills him (i.e. resembling a 'hot potato' game). Fig. 8 presents *Shift Adapter* game scheme.

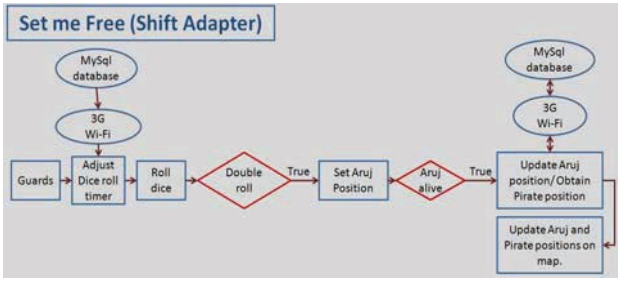

Fig. 8. The Set me Free!'s Guards Shift Adapter functional scheme.

To encourage the *Guards* follow some short of strategy in Aruj's relocations (thereby making the game more interesting), the last known position of the outdoors *Pirate* player is revealed at the time they decide for the next Aruj position. The Aruj AR marker's position is updated every time the *Guards* set one new position for him. Fig. 9 presents the *Set Me Free!* AR application interface, as seen by the *Pirate* player.

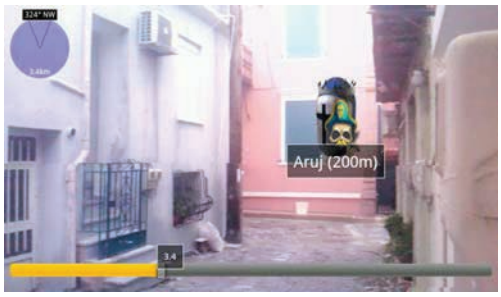

Fig. 9. Augmented Reality view of Aruj's position.

Fig. 10 presents the *Set me Free!* Pirate AR game functional scheme.

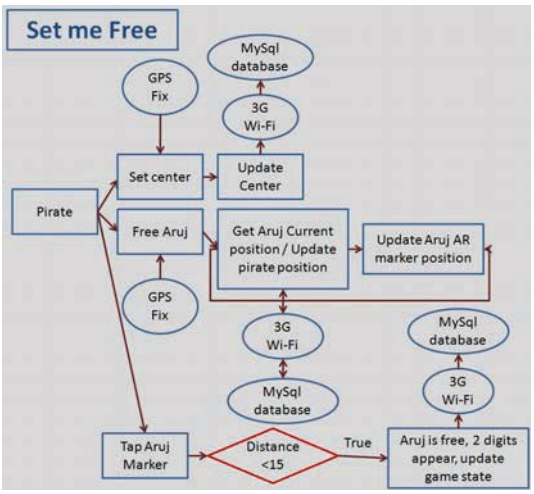

Fig. 10. The Set me Free! Pirate AR game functional scheme.

The game ends when the outdoor player manages to approach and release Aruj (the *Guards* are notified about this incident). Upon releasing Aruj, the *Pirate* receives a 2 digit part of the unlocking code.

4. EVALUATION

4.1 Evaluation methodology

In order to thoroughly evaluate Barbarossa we have conducted fully open user trials. The process involved raising awareness about the game by announcing the trial dates via various communication channels (flyers, mailing lists, game-relevant websites, social media, etc). To collect evaluation data we edited separate questionnaires per phase and player role; the questionnaires have been largely prepared based on the legacy of game evaluation guidelines proposed in the literature [27, 36-39] and further inspired by questionnaires and interviews used in other pervasive game prototype trials [6, 11-13, 17-20, 26, 40]. Having completed the game, all the three players of each team filled in a questionnaire and replied to some interview questions tailored to their role. Moreover players' answers have been crosschecked with detailed log data maintained for each player throughout the game sessions in the backend system of *Barbarossa*.

4.2 Evaluation results

4.2.1 Game pace player attitudes

In the *Treasure Hunt* scenario the average duration of the game sessions has been 12 hours and 5 minutes (time between the scanning of the first and last clue). The above data along with the *Treasure Hunters* response to the interview question about the effort they put (8 out of 10 players claimed they put medium-level effort) confirmed that the design goals of *Treasure Hunt* (i.e. to function as prolonged and restrained effort game scenarios) have been actually achieved.

As regards the second phase of *Barbarossa*, the *Knight* players have had opposing opinions with respect to the extent to which they increased their effort due to not having the option to pause and resume the game (6 players agreed with the above statement, 3 were neutral and 1 disagreed). All the *Pirate* players though,

who were also unable to pause and resume the game during the *Set Me Free!* game sessions, reported that this feature made the game more interesting and fun (4 strongly agreed with the above statement and 6 agreed).

Most *Knight* and *Pirate* players confirmed they put considerable effort to complete their game goals (18 players claimed to put high effort, 1 player claimed to put medium effort and 1 player low effort). Indeed, the logged data provided evidence for short time game sessions with no interruptions or errors, high effort involvement and relatively quick completion of game goals by starting a small number of game sessions.

The average sessions started by the ten players into the *ManHunt* scenario were two, the average duration of each session was 3'40'' and the average distance travelled by each player along each session was 261 meters. The answers of the *Knight* players and the log data verified that they completed the game within their first try.

In the Set Me Free! scenario all 10 players but one, who interrupted the game to ask the developers to explain a minor feature of the game, also completed their game goal on a single game session, without experiencing any kind of errors or interruptions. The average duration of the game sessions has been 5'53'' and the average distance the players covered during those sessions was 363 meters. Players effort along those sessions has been high as also indicated by the 11, on average, relocations of Aruj position.

4.2.2 User-generated content & Orchestration

Moreover, all *Treasure Hunters* expressed appreciation for the fact that the game was staged around their own area and that they could play there anywhere and at any time. Finally, all *Pirate* players testified that being aware that their actions affected the Guards play and vice versa, was a quite interesting game feature revealing that user-generated game content may serve as a useful and effective tool to seamlessly integrate the orchestration actions needed with the actual game fulfilling this way its orchestration requirements.

4.2.3 Playing anytime, anywhere

All (but one) *Treasure Hunters* and the *Knights* agreed that being able to play anytime, anywhere was rather convenient and useful. The evaluators particularly enjoyed the feeling of being not confined in a predefined game area and thinking freely when considering the possible locations of clues and the chest. Moreover, all the *Pirate* players (but one who was neutral) confirmed that being allowed to decide upon the time and place to play (in consultation with the developers) was convenient and useful too. The logged data verify that *Knight* and *Pirate* players' game sessions were staged across different locations and dates/times.

4.2.4 Communication

Most players have chosen Facebook group chat to communicate with their teammates as well as the developers. Standard phone calls have also been used for the communication between the *Treasure Hunters* and the *Pirates*. Generally, letting players choose their own communication method has been received positively as all the players agreed that the communication with other players was straightforward.

4.2.5 Localization and communication uncertainty

As regards all the game scenarios, there were no errors captured. The user trials of *Barbarossa* revealed that designing prolonged game sessions which require restrained player effort and involve moderate usage of technology, and short time scale game sessions, which require higher player effort with intense use of technology helped into reducing game interruptions caused by the use of technology reducing the uncertainty significantly.

5. CONCLUSION AND FUTURE WORK

This article introduced *Barbarossa*, an outdoors role-playing pervasive game based upon open and portable design architecture. *Barbarossa* eases the relocation of the game at any urban setting, simplifies the game setup and prevents the players' frustration typically experienced due to human orchestration.

User trials evaluation results (compiled from questionnaires, interviews and logged data) provide sufficient evidence as regards the accomplishment of our game design goals and largely verify the research contributions achieved on the grounds of *Barbarossa*. The main conclusions drawn from user trials are summarized in the following:

- Designing games with respect to the game session player effort requirements and duration in proportion with the technologies used may reduce the number of errors and interruptions during a game session reducing that way the uncertainty of technology usage and preserving the players immersion.
- The ability of pausing/resuming the game may be a convenient and useful instrument for players, although -depending on the game scenario- the lack of this feature may increase players' commitment and enjoyment.
- Being able to unfold the game scenario anytime, anywhere has been highly appreciated by players. Even though the game is staged on a certain area and cannot be relocated by the players elsewhere, the feeling of not being spatially confined has been rather appealing.
- User-generated content can serve both as a main ingredient that increases the game's fun factor and as an effective technique to take care of orchestration actions needed.

A fully functional prototype of Barbarossa, as described in this paper, is already at hand (see http://www.barbarossarpg.com/). In the future, we plan to develop a fully-automated orchestration engine as regards the clues delivery in the context of the *Treasure Hunt* scenario, using photos of specific locations retrieved from the Flickr [41] or the Google Places [32] APIs. Moreover we plan to develop a fully portable version of the *ManHunt* game using the Google Places API [34] and automatically generated starting and ending locations for Ilyas, so as to support every possible environment and allow players outside the city of Mytilene to participate in second phase of the game. Finally, we plan to perform more extensive game trials on *Barbarossa* to validate our initial evaluation results.

6. REFERENCES

[1] S. Benford, C. Magerkurth, P. Ljungstrand, Bridging the physical and digital in pervasive gaming, *Communications of the ACM*, 48 (2005) 54-57.

[2] V. Kasapakis, D. Gavalas, Design Aspects and Context-Awareness in Pervasive Games, in: B. Guo, D. Riboni, P. Hu (Eds.) *Creating Personal, Social, and Urban Awareness through Pervasive Computing*, IGI Global, 2013, pp. 131-156.

[3] B.K. Walther, Atomic Actions -- Molecular Experience: Theory of Pervasive Gaming, *Computers in Entertainment*, 3 (2005) 1-13.

[4] PK, Parallel Kingdom, 2013.

[5] J. Peitz, S. Björk, INSECTOPIA Using the Real World as a Game Resource, *Space Time Play*. Birkhäuser, Basel, 2007, pp. 294-295.

[6] I. Chatzigiannakis, G. Mylonas, O. Akribopoulos, M. Logaras, P. Kokkinos, P. Spirakis, The "Hot Potato" Case: Challenges in Multiplayer Pervasive Games Based on AdHoc Mobile Sensor Networks and the Experimental Evaluation of a Prototype Game, *CoRR abs/1002.1099*, (2010).

[7] S. Bjrk, J. Falk, R. Hansson, P. Ljungstrand, Pirates! - Using the Physical World as a Game Board, *Interact IFIP TC.13 Conference on Human-Computer Interaction*, 2001.

[8] S. Benford, D. Rowland, M. Flintham, R. Hull, J. Reid, J. Morrison, K. Facer, B. Clayton, "Savannah": Designing a Location Based Game Simulating Lion Behaviour, *International Conference on Advances in Computer Entertainment Technology*, 2004.

[9] Hackolog, Arduino Geo-Location Game Known as Urban Defender, in: *Hackolog (Ed.)*, 2013.

[10] P. Kiefer, S. Matyas, C. Schlieder, Systematically Exploring the Design Space of Location-based Games, 4*th International Conference on Pervasive Computing* (Pervasive'2006), 2006, pp. 183-190.

[11] A.D. Cheok, A. Sreekumar, C. Lei, L.M. Thang, Capture the Flag: Mixed-Reality Social Gaming with Smart Phones, *IEEE Pervasive Computing*, 5 (2006) 62-63.

[12] S. Benford, M. Flintham, A. Drozd, R. Anastasi, D. Rowland, N. Tandavanitj, M. Adams, J. Row-Farr, A. Oldroyd, J. Sutton, Uncle Roy All Around You: Implicating the City in a Location-Based Performance, *ACM Advanced Computer Entertainment*, (2004).

[13] S. Benford, A. Crabtree, M. Flintham, A. Drozd, R. Anastasi, M. Paxton, Can You See Me Now?, *ACM Transactions on Computer-Human Interaction*, 13 (2006).

[14] S. Olli, All The World's A Botfighter Stage: Notes on Location-based Multi-User Gaming, *Computer Games and Digital Cultures Conference*, 2002, pp. 35-44.

[15] C.A. Lindley, Game space design foundations for trans-reality games, *ACM SIGCHI International Conference on Advances in computer entertainment technology*, ACM, Valencia, Spain, 2005, pp. 397-404.

[16] C.A. Lindley, Trans-Reality Gaming, 2*nd Annual International Workshop in Computer Game Design and Technology*, 2004.

[17] A.D. Cheok, E.T. Khoo, Age Invaders: Inter-generational Mixed Reality Family Game, *The International Journal of Virtual Reality*, 5 (2006) 45-50.

[18] W. Wetzel, L. Blum, R. McCall, L. Oppermann, T.S. Broeke, Z. Szalavári, Final Prototype of TimeWarp application, *IPCity*, 2009.

[19] B. Guo, R. Fujimura, D. Zhang, M. Imai, Design-in-play: improving the variability of indoor pervasive games, *Multimedia Tools Appl.*, 59 (2012) 259-277.

[20] J. Fischer, I. Lindt, J. Stenros, Final Crossmedia Report (part II) – Epidemic Menace II Evaluation Report, *Integrated Project on Pervasive Gaming*, 2006.

[21] A.D. Cheok, K.H. Goh, W. Liu, F. Farbiz, S.W. Fong, Human Pacman: A Mobile, Wide-Area Entertainment System Based on Physical, Social, and Ubiquitous Computing, *Personal and Ubiquitous Computing*, 8 (2004) 71-81.

[22] W. Piekarski, B. Thomas, ARQuake: the outdoor augmented reality gaming system, *Commun. ACM*, 45 (2002) 36-38.

[23] I. Herbst, A.-K. Braun, R. McCall, W. Broll, TimeWarp: Interactive Time Travel with a Mobile Mixed Reality Game, *10th International Conference on Human Computer Interaction with Mobile Devices and Services*, ACM, 2008.

[24] M. Flintham, R. Anastasi, S. Benford, A. Drozd, J. Mathrick, D. Rowland, N. Tandavanitj, M. Adams, J. Row-Farr, A. Oldroyd, J. Sutton, Uncle Roy All Around You: Mixing Games and Theatre on the City Streets, *1st International Conference on Digital Games Research Association* (DIGRA'03), 2003.

[25] S. Hannamari, J. Kuittinen, M. Montola, Insectopia Evaluation Report, *Integrated Project on Pervasive Gaming*, 2007.

[26] C. Linehan, B. Kirman, S. Lawson, M. Doughty, Blowtooth: pervasive gaming in unique and challenging environments, *CHI '10 Extended Abstracts on Human Factors in Computing Systems*, ACM, Atlanta, Georgia, USA, 2010, pp. 2695-2704.

[27] K. Jegers, Pervasive Game Flow: Understanding Player Enjoyment in Pervasive Gaming, *Computers in Entertainment*, 5 (2007).

[28] M. Montola, J. Stenros, A. Waern, *Pervasive Games: Theory and Design*, Morgan Kaufmann Publishers Inc.2009.

[29] A. Konstam, *Piracy: The Complete History*, Osprey Publishing2008.

[30] D. Wave, Multi-format 1D/2D barcode image processing library with clients for Android, Java, 2013.

[31] SunSPOT, Sun SPOT World Home, 2011.

[32] Google, Getting Started - Google Places API 2013.

[33] Google, The Google Directions API - Google Maps API Web Services — Google Developers, 2013.

[34] Yahoo!, Yahoo! Weather - Yahoo! Developer Network, 2013.

[35] J. Wetherell, android-augment-reality-framework A framework for creating augmented reality Apps on Android, 2013.

[36] T. Fullerton, C. Swain, S. Hoffman, *Game Design Workshop: Designing, Prototyping, and Playtesting Games*, CMP Books2004.

[37] H. Korhonen, E.M.I. Koivisto, Playability heuristics for mobile games, *Proceedings of the 8th conference on Human-computer interaction with mobile devices and services*, ACM, Helsinki, Finland, 2006, pp. 9-16.

[38] C. Jennett, A.L. Cox, P. Cairns, S. Dhoparee, A. Epps, T. Tijs, A. Walton, Measuring and defining the experience of immersion in games, *Int. J. Hum.-Comput. Stud.*, 66 (2008) 641-661.

[39] H. Saarenpää, Data Gathering Methods for Evaluating Playability of Pervasive Mobile Games, (2008).

[40] R. Zender, R. Metzler, U. Lucke, FreshUP—A pervasive educational game for freshmen, *Pervasive and Mobile Computing*, (2013).

[41] Flickr, Flickr Services, 2013.